Delft University of Technology

Fac. of Information Technology and Systems **Control Systems Engineering**

Technical report bds:00-17

Model predictive control for railway networks*

B. De Schutter and T. van den Boom

If you want to cite this report, please use the following reference instead:
B. De Schutter and T. van den Boom, "Model predictive control for railway networks," *Proceedings of the 2001 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM'01)*, Como, Italy, pp. 105–110, July 2001.

Control Systems Engineering
Faculty of Information Technology and Systems
Delft University of Technology
Delft, The Netherlands
Current URL: https://www.dcsc.tudelft.nl

Model predictive control for railway networks

B. De Schutter and T. van den Boom

Delft University of Technology
Control Lab, Faculty of Information Technology and Systems
P.O. Box 5031, 2600 GA Delft, The Netherlands
{b.deschutter,t.j.j.vandenboom}@its.tudelft.nl
http://lcewww.et.tudelft.nl

Abstract— **Model predictive control (MPC) is a very popular controller design method in the process industry. Usually MPC uses linear discrete-time models. In this paper we extend MPC to a class of discrete-event systems with both hard and soft synchronization constraints. Typical examples of such systems are railway networks, subway networks, and other logistic operations. In general the MPC control design problem for these systems leads to a nonlinear non-convex optimization problem. We also show that the optimal MPC strategy can be computed using an extended linear complementarity problem.**

I. Introduction

We present a model predictive control (MPC) framework for a special class of discrete event systems, namely railway and subway networks.

MPC [1], [2], [3] is a very popular control design technique in the process industry. An important advantage of MPC is that it allows the inclusion of constraints on the inputs and outputs, and that it can handle changes in the system parameters by using a moving horizon approach, in which the model and the control strategy are continuously updated. Conventional MPC uses discrete-time models (i.e., models consisting of a system of difference equations). In [4] we have extended MPC to a special class of discrete event systems.

Typical examples of discrete event systems are flexible manufacturing systems, telecommunication networks, parallel processing systems, traffic control systems, and logistic systems. The class of discrete event systems essentially consists of man-made systems that contain a finite number of resources (such as machines, communications channels, or processors) that are shared by several users (such as product types, information packets, or jobs) all of which contribute to the achievement of some common goal (the assembly of products, the end-to-end transmission of a set of information packets, or a parallel computation) [5]. In general, models that describe the behavior of a discrete event system are nonlinear in conventional algebra. However, there is a class of discrete event systems – the max-plus-linear discrete event systems – that can be described by a model that is "linear" in the max-plus algebra [5], [6], which has maximization and addition as its basic operations. The max-plus-linear discrete event systems can be characterized as the class of discrete event systems in which only synchronization and no concurrency or choice occurs. So typical examples are serial production lines, production systems with a fixed routing schedule, and railway networks with rigid connection constraints.

In [4] we have extended MPC to the class of max-plus-linear discrete event systems. The synchronization constraints for max-plus-linear discrete event systems are "hard", i.e., the constraints should always be met. In this paper we further extend the MPC framework to a class of discrete event systems with both *soft* and *hard* synchronization constraints, i.e., in some cases we allow an event to start although not all pre-scheduled predecessor events have been completed, but at a cost. This could occur in a logistics context or a railway operations context, where a train should give pre-defined connections to other trains. However, if some of these trains have a too large delay, then it is sometimes better — from a global performance viewpoint — to let the train depart anyway in order to prevent an accumulation of delays in the network. Of course, missed connections lead to a penalty due to dissatisfied passengers. Note that in [4] we have only considered hard synchronization constraints. Other work in connection with the modeling and control of railway networks (mainly in a discrete event context) can be found in [7], [8], [9], [10], [11], [12].

In this paper we will first derive a model for a railway system with hard and soft synchronization constraints. Next, we will define a control design problem for such a system where we can break a connection if delays occur and if this leads to a better global performance. We use an MPC approach (which has the following ingredients: a prediction horizon, a receding horizon procedure, and a regular update of the model and re-computation of the optimal control). In general this will lead to a hard non-convex nonlinear optimization problem. However, we will show that the trajectories of the system can be described by an extended linear complementarity problem (ELCP) [13], for which we can compute a parameterized solution [13]. Afterwards, we can then compute the optimal control over this solution set. The advantage is that we now have to solve a sequence of optimization problems with a convex feasible set (although the objective function is still nonlinear and non-convex). Computational experiments show that (for small sized problems or for a small control horizon) the ELCP approach is much faster and yields a better minimum than the straightforward nonlinear optimization approach.

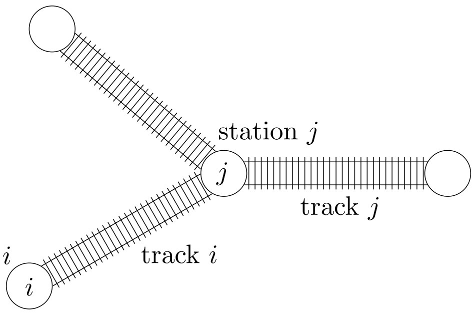

Fig. 1. A part of a railway network.

II. Model

Consider a railway operations system. The nominal operation of the system follows a time schedule with a period T. We assume that all the trains follow a pre-scheduled route. Let n be the number of tracks in the network. Each track of the railway network has a number and a virtual train allocated to it. For the sake of simplicity we will say "(virtual) train j" to denote the (physical) train on track j, and "station j" to denote the station at the beginning of track j (cf. Figure 1). Let $x_j(k)$ be the time instant at which train j departs from station j for the kth time. Let $d_j(k)$ be the departure time for this train according to the time schedule.

Let $\mathcal{C}_j(k)$ be the set of trains to which the kth train on track j gives a connection. This set can be divided in a set of hard connections $\mathcal{C}_j^{\text{hard}}(k)$ (e.g., if the train on track i and the train on track j are physically the same train, or if it is a very important connection that should be guaranteed at all cost) and a set of soft connections $\mathcal{C}_j^{\text{soft}}(k)$ (e.g., local trains to which the train j should give connection, but if the local train $i \in \mathcal{C}_j^{\text{soft}}(k)$ has a too large delay, then the connection may be broken; however, in that case a cost $c_{ij}^{\text{broken}}(k)$ is associated with the broken connection (see (2) and Remark 1 for a refinement)). We have $\mathcal{C}_j^{\text{hard}}(k) \cap \mathcal{C}_j^{\text{soft}}(k) = \emptyset$ and $\mathcal{C}_j^{\text{hard}}(k) \cup \mathcal{C}_j^{\text{soft}}(k) = \mathcal{C}_j(k)$.

Let $a_{ij}(k)$ be the travel time from station i to station j for each train $i \in \mathcal{C}_j(k)$. Furthermore, we define a minimum connection time $t_{ij}^{\min}(k)$ for passengers to get from train i to train j for each train $i \in \mathcal{C}_j(k)$ (if virtual trains i and j are physically the same train, then this time corresponds to the minimum stopping time of train j at station j to allow passenger to get off or on the train).

Now we have the following constraints for the kth departure time $x_j(k)$ of train j:

- the time schedule constraint:

$$x_j(k) \geqslant d_j(k) \ .$$

- hard synchronization constraint:

$$x_j(k) \geqslant x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) \qquad \text{for each } i \in \mathcal{C}_j^{\text{hard}}(k),$$

where the notation 1_{ij}^* is used to denote 1 if the kth train j gives connection to the $(k-1)$th train i, and 0 if the kth train j gives connection to the kth train i (and if some trips last longer than the twice the cycle time T of the schedule, 1_{ij}^* might be equal to 2, and so on — see also the example in Section V). Note that in general 1_{ij}^* might even also be a function of k. However, for the sake of simplicity, we will only consider constant 1_{ij}^*'s in this paper.

- soft synchronization constraint:

For each train $i \in \mathcal{C}_j^{\text{soft}}(k)$ we have

$$x_j(k) \geqslant x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) \qquad \text{if the connection takes place,}$$

$$x_j(k) < x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) \qquad \text{if the connection is broken.}$$

If we introduce a control variable $u_{ij}(k) \geqslant 0$, then we can combine these equations into

$$x_j(k) \geqslant x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) - u_{ij}(k)$$

where $u_{ij}(k)$ can be used to guarantee or to break a connection.

Since we let a train depart as soon as all connection conditions are satisfied, we have

$$\begin{aligned} x_j(k) = \max\big(& d_j(k), \\ & \max_{i \in \mathcal{C}_j^{\text{hard}}(k)} (x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k)), \\ & \max_{i \in \mathcal{C}_j^{\text{soft}}(k)} (x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) - u_{ij}(k))\big) \end{aligned} \tag{1}$$

Note that in a nominal, well-defined time schedule the term $d_j(k)$ in (1) will be the largest. However, if due to unforeseen circumstances (an incident, a late departure, etc.) train i has a delay the corresponding term can become larger than the others.

Define $t_{ij}^{\text{slack}}(k)$ as the slack time[1] of the arrival of train $i \in \mathcal{C}_j^{\text{soft}}(k)$ at station j (transit time $t_{ij}^{\min}(k)$ included) w.r.t. the actual kth departure time of train j:

$$t_{ij}^{\text{slack}}(k) = x_i(k - 1_{ij}^*) + a_{ij}(k) + t_{ij}^{\min}(k) - x_j(k) \ .$$

If $t_{ij}^{\text{slack}}(k) \leqslant 0$ then the connection is completely guaranteed (with enough time for the passengers to change trains). If $t_{ij}^{\text{slack}}(k) > t_{ij}^{\min}(k)$ then train j leaves the station before the arrival of train i. If $0 < t_{ij}^{\text{slack}}(k) \leqslant t_{ij}^{\min}(k)$ then the connection is guaranteed partly (i.e., fast-running passengers can get the connection, but slower ones may lose it). Therefore, we define the cost of a broken connection as the

[1] Note that this slack time is a function of the control variable $u_{ij}(k)$ via $x_j(k)$.

piecewise-linear function

$$J_{\text{broken}}(t^{\text{slack}}, t^{\min}, c^{\text{broken}}) = \begin{cases} 0 & \text{if } t^{\text{slack}} \leqslant 0, \\ \dfrac{c^{\text{broken}}}{t^{\min}}\, t^{\text{slack}} & \text{if } 0 < t^{\text{slack}} \leqslant t^{\min}, \\ c^{\text{broken}} & \text{if } t^{\text{slack}} > t^{\min}. \end{cases} \quad (2)$$

III. The railway MPC problem

We define the following cost function over a given prediction horizon N_{p}:

$$J_{\text{cost}}(k) = \sum_{l=0}^{N_{\text{p}}-1} \sum_{j=1}^{n} |x_j(k+l) - d_j(k+l)| + \lambda \sum_{l=0}^{N_{\text{p}}-1} \sum_{j=1}^{n} \sum_{i \in \mathcal{C}_j^{\text{soft}}(k+l)} J_{\text{broken}}(t_{ij}^{\text{slack}}(k+l), t_{ij}^{\min}(k+l), c_{ij}^{\text{broken}}(k+l))$$

where $\lambda > 0$ is a weighting factor. This cost function has two components: the first tries to keep the trains running on schedule, whereas the second penalizes broken connections. The factor λ determines the trade-off or relative weight of the two components of the MPC cost function.

Now we consider the following controller design problem — which will be called the *railway MPC problem* at cycle k:

$$\min_{u_{ij}(k),\ldots,u_{ij}(k+N_{\text{p}}-1)} J_{\text{cost}}(k)$$

subject to (1) and $u_{ij}(k+l) \geqslant 0$ for all i, j and $l = 0, \ldots, N_{\text{p}}-1$.

In addition, to reduce the number of control variables we can — just as in conventional MPC — introduce a control horizon N_{c} ($\leqslant N_{\text{p}}$) and set

$$u_{ij}(k+l) = u_{ij}(k+N_{\text{c}}-1) \quad \text{for } l = N_{\text{c}}, \ldots, N_{\text{p}}-1\,. \quad (3)$$

This condition can be interpreted as follows: if after N_{c} steps the delays have died out (i.e., it is not necessary to break connections anymore or equivalently, $u_{ij}(k+N_{\text{c}}) = 0$ for all i, j), then we do not break any connections in the subsequent steps either. On the other hand, if the delays are still such that a connection should be broken in step $k + N_{\text{c}}$, then we will also break these connections in the subsequent steps[2].

Just like in conventional MPC we use a moving horizon approach, i.e., the railway MPC problem is solved for each cycle, then the computed controls for that cycle are applied, and meanwhile the model is updated, and the computation is performed again for the next cycle. This implies that we can also include predictable future delays (due to incidents, broken power lines, works, ...) into our prediction model.

Note that the parameter N_{p} should be chosen such that it covers the (expected) period over which the delays will die out. The choice of N_{c} mainly depends on the computational complexity of the problem. For small-sized networks we can take N_{c} rather large, whereas for large networks a small N_{c} will be necessary to be able to compute the MPC solution sufficiently fast (i.e., before the start of the next cycle of the railway network).

In general each step of the railway MPC problem leads to a non-convex nonlinear optimization problem. This problem can be solved using, e.g., a multi-start local optimization method such as multi-start sequential quadratic programming. Also note that the feasible set of the railway MPC problem is non-convex since (1) is non-convex. In the next section we will present an alternative approach to compute the optimal MPC control input which is based on a mathematical programming problem called the extended linear complementarity problem.

IV. Link with the ELCP

The Extended Linear Complementarity Problem (ELCP) is defined as follows [13]:
Given $A \in \mathbb{R}^{p \times n}$, $B \in \mathbb{R}^{q \times n}$, $c \in \mathbb{R}^p$, $d \in \mathbb{R}^q$ and m subsets $\phi_1, \ldots, \phi_m$ of $\{1, \ldots, p\}$, find $z \in \mathbb{R}^n$ such that

$$\prod_{i \in \phi_j} (Az - c)_i = 0 \quad \text{for } j = 1, \ldots, m, \quad (4)$$

subject to

$$Az \geqslant c \quad (5)$$

$$Bz = d\ . \quad (6)$$

Equation (4) represents the complementarity condition of the ELCP. One possible interpretation of this condition is the following: each set ϕ_j corresponds to a group of inequalities of $Az \geqslant c$ and in each group at least one inequality should hold with equality, i.e., the corresponding residue should be equal to 0. So for each j there should exist an index $i \in \phi_j$ such that $(Az - c)_i = 0$.

The formulation of the ELCP arose from our research on nonlinear resistive networks, discrete event systems (max-plus-linear systems, applications in max-plus algebra, and min-max-plus systems) and hybrid systems (traffic signal control, and first-order hybrid systems with saturation).

Let us now show that the evolution equations and the constraints of the MPC optimization problem considered in the previous section can be recast as an ELCP. Clearly, the non-negativity constraint on $u_{ij}(k)$ and the control horizon constraint fit the ELCP framework. Now we show that (1) can also be written as an ELCP. This will be done by showing that an expression of the form $x_j(k) = \max(d_j(k), e_j(k), f_j(k))$ is an ELCP. If we then add the conditions that $e_j(k)$ and $f_j(k)$ should be equal to the second term and the third term of the right-hand side of (1) and if we take into account that the merge of two ELCPs is

[2] We can even take the decrease or growth of the delays into account by using a constant growth/decrease rate condition of the form $\Delta u_{ij}(k+l) = \Delta u_{ij}(k+N_{\text{c}}-1)$ for $l = N_{\text{c}}, \ldots, N_{\text{p}}-1$ instead of (3) where $\Delta s(k) = s(k) - s(k-1)$.

also an ELCP, we have recursively shown that (1) can also be written as an ELCP. The condition[3] $x_j = \max(d_j, e_j, f_j)$ can be rewritten as $x_j - d_j \geqslant 0$, $x_j - e_j \geqslant 0$, $x_j - f_j \geqslant 0$, with $x_j = d_j$ or $x_j = e_j$ or $x_j = f_j$. The latter condition can be rewritten as $(x_j - d_j)(x_j - e_j)(x_j - f_j) = 0$. Hence, we have obtained an ELCP. As a consequence, the trajectories of the railway system can be described by an ELCP.

The solution set of an ELCP is the union of a subset of faces of the polyhedron defined by $Az \leqslant c$, $Bz = d$. So it is the union of convex sets. In [13] we have developed an algorithm that yields a parametric description of the solution set of an ELCP in which each face is presented by its vertices. More specifically, the solution set of the ELCP (4)–(6) is characterized by a set of vectors $V = \{z^i \,|\, i = 1, \ldots, r\}$ and a set of index sets $\Lambda = \{\psi_j \,|\, j = 1, \ldots, s\}$ such that for any j any convex combination of the form

$$\sum_{i \in \psi_j} \nu_i z^i \qquad \text{with } \nu_i \geqslant 0 \text{ and } \sum_{i \in \phi_j} \nu_i = 1$$

is a solution of (4)–(6). The vectors of V correspond to vertices of the polyhedron defined by the system (5)–(6) and each index set ψ_j corresponds to a face of this polyhedron. The optimal MPC strategy can now be obtained by determining for each index set ψ_j the combination of the ν_i's for which the objective function $J_{\mathrm{cost}}(k)$ reaches a global minimum (note that this is an optimization over a convex set) and afterwards selecting the overall minimum.

The advantage of this approach compared to straightforward nonlinear constrained optimization of the railway MPC problem is that in the ELCP approach we have to solve a sequence of optimization problems with a convex feasible set instead of one big problem with a non-convex feasible set. Optimization problems with a convex feasible set (albeit with a non-convex objective function) are easier to solve numerically than problems with a non-convex feasible set. Note however that the algorithm of [13] to compute the solution set of a general ELCP requires exponential execution times, which that the ELCP approach is not feasible if N_{c} is large.

Our computational experiments have shown that in most cases the determination of the minimum value of the objective functions given above is a well-behaved problem in the sense that using a local minimization routine (that uses, e.g., sequential quadratic programming) starting from different initial points almost always yields the same numerical result (within a certain tolerance). So (for small sized problems or for a small control horizon) the ELCP approach is much faster and yields a better minimum than the straightforward nonlinear optimization approach.

Remark 1 To get a smoother optimization problem we can introduce another, smoother cost function such as, e.g.,

$$J^*_{\mathrm{broken}}(t^{\mathrm{slack}}, t^{\min}, c^{\mathrm{broken}}) = \begin{cases} 0 & \text{if } t^{\mathrm{slack}} \leqslant 0 \\ \dfrac{c^{\mathrm{broken}}}{2\gamma}\Big(1 + \tanh\Big(\alpha\Big(\dfrac{t^{\mathrm{slack}}}{t^{\min}} - \gamma\Big)\Big)\Big) \cdot \dfrac{t^{\mathrm{slack}}}{t^{\min}} & \\ & \text{if } 0 < \dfrac{t^{\mathrm{slack}}}{t^{\min}} \leqslant \gamma, \\ \dfrac{c^{\mathrm{broken}}}{2\gamma}\Big(1 + \tanh\Big(\alpha\Big(\dfrac{t^{\mathrm{slack}}}{t^{\min}} - \gamma\Big)\Big)\Big) \cdot & \\ \qquad \Big(\gamma + \beta\Big(\dfrac{t^{\mathrm{slack}}}{t^{\min}} - \gamma\Big)\Big) & \text{if } \dfrac{t^{\mathrm{slack}}}{t^{\min}} > \gamma. \end{cases} \tag{7}$$

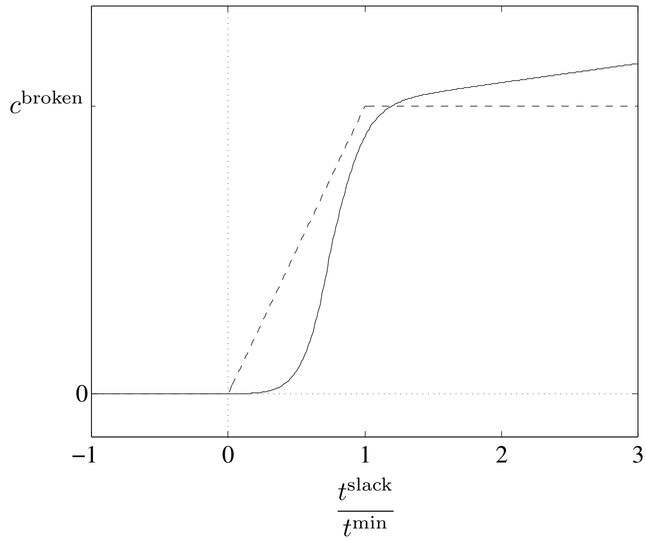

Fig. 2. The piecewise-linear cost function J_{broken} (dashed line) and the smoother cost function J^*_{broken} (full line) (with $\alpha = 4$, $\beta = 0.05$ and $\gamma = 0.75$).

The graph of this function is given in Figure 2. The cost function J^*_{broken} also better corresponds to what we expect in reality instead of the piecewise-linear cost function J_{broken} defined by (2).

Furthermore, if $t^{\mathrm{slack}}_{ij}(k)$ is nonpositive (or if there is another index i' such that $t^{\mathrm{slack}}_{ij}(k) > t^{\mathrm{slack}}_{i'j}(k)$), then $u_{ij}(k)$ does not influence the value of the objective function anymore. Therefore, we add an extra term of the form

$$\eta \sum_{l=1}^{N_{\mathrm{p}}-1} \sum_{j=1}^{n} \sum_{i \in \mathcal{C}^{\mathrm{soft}}_j(k+l)} u_{ij}(k+l) \tag{8}$$

to the MPC cost function. In that way, we get the smallest possible values of $u_{ij}(k)$. This also enables us to see more clearly which connections are broken or not. ◇

V. Worked example

Consider the railroad network of Figure 3. There are 4 stations in this railroad network (A, B, C and D) that are connected by 6 single tracks. There are two trains available.

[3]For ease of notation we drop the index k.

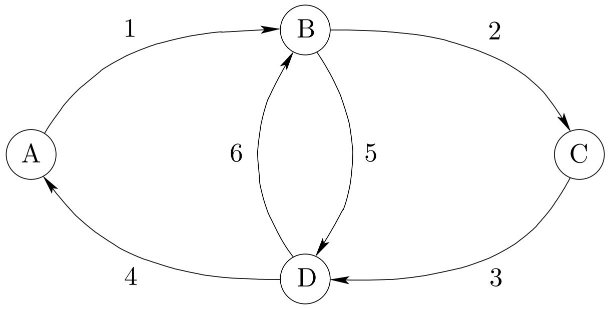

Fig. 3. The railroad network of the example of Section V.

TABLE I
THE NOMINAL TRAVELING TIMES AND THE DEPARTURE TIMES FOR THE RAILROAD NETWORK OF THE EXAMPLE OF SECTION V.

Track	From station	To station	Nominal traveling time	Scheduled departure time modulo 60
1	A	B	15	00
2	B	C	9	18
3	C	D	10	30
4	D	A	11	45
5	B	D	22	20
6	D	B	21	50

The first train follows the route A $\to$ B $\to$ C $\to$ D $\to$ A and the second train follows the route B $\to$ D $\to$ B. We assume that there exists a periodic timetable that schedules the earliest departure times of the trains. The period of the timetable is $T = 60$ minutes. So if a departure of a train from station B is scheduled at 5.18 a.m., then there is also scheduled a departure of a train from station B at 6.18 a.m., 7.18 a.m., and so on. Table I summarizes the information in connection with the nominal traveling times and the departure times. All the times are measured in minutes. The indicated departure times are the earliest departure times in the initial station of the track expressed in minutes after the hour. The first period starts at time $t = 0$. At the beginning of the first period the first train is in station A and the second train is in station B.

Suppose that we have to guarantee the following connections in order to allow the passengers to change trains:

- the train on track 2 has to wait for the train on track 6,
- the train on track 4 has to wait for the train on track 5,
- the train on track 5 has to wait for the train on track 1,
- the train on track 6 has to wait for the train on track 3.

These connections are *soft* constraints. The hard connection constraints are that the trains on tracks 1, 2, 3 and 4 are physically the same train, and the same holds for the trains on tracks 5 and 6. The passengers get 2 minutes to change trains (for soft connections) and 1 minute to get out of the train (for hard connections).

Each train departs as soon as all the connections are guaranteed (except for a soft connection when it is broken), the passengers have gotten the opportunity to change over and the earliest departure time indicated in the timetable has passed. We assume that in the first period all the trains depart according to schedule. Recall that $x_j(k)$ is the time instant at which the train on track j departs from the initial station of the track for the kth time.

Now we write down the equations that describe the evolution of the $x_j(k)$'s. First we consider the train on track 1 and we determine $x_1(k)$, the time instant at which this train departs from station A for the kth time. At the beginning of the first period the train is in station A. So if k is equal to 1, the train departs from station A at time $t = 0$. If k is greater than 1, the train departs from station A for the kth time as soon as it has arrived in station A for the $(k-1)$th time[4] and the passengers have got the time to get out of the train and the earliest departure time indicated in the timetable has passed. The train arrives in station A for the $(k-1)$th time at time instant $x_4(k-1)+a_{41}(k)$, and afterwards, the passengers have $t_{41}^{\min}(k) = 1$ minute to get out of the train. Since the system operates under a periodic timetable with period T, the kth departure time of the train on track 1 according to the timetable is $0 + (k-1)T$. So if we set[5] $x_4(0) = -\infty$, then we have

$$x_1(k) = \max(x_4(k-1) + a_{41}(k) + 1,\, 0 + (k-1)T)$$

for $k = 1, 2, \ldots$

The train on track 1 will arrive for the kth time in station B at time instant $x_1(k)+a_{12}(k)$, after which the passengers have $t_{12}^{\min}(k) = 1$ minute to get out of the train. If k is greater than 1, the train has to wait for the passengers of the train on track 6, which arrives in station B at time instant $x_6(k-1) + a_{62}(k)$. The passengers have $t_{62}^{\min}(k) = 2$ minutes to change trains. According to the timetable the train on track 2 can only depart after time instant $18+(k-1)T$. Furthermore, since the connection constraint is soft, we introduce a control variable $u_{62}(k)$ to break the connection if necessary. Hence, we have

$$\begin{aligned} x_2(k) = \max(&x_1(k) + a_{12}(k) + 1, \\ &x_6(k-1) + a_{62}(k) + 2 - u_{62}(k), 18 + (k-1)T) \end{aligned}$$

for $k = 1, 2, \ldots$ with $x_6(0) = -\infty$. Note that — referring to (1) — we have $1^*_{12}(k) = 0$ since the kth train on track 2 is the same train as the kth train on track 1, and $1^*_{62}(k) = 1$ since the kth train on track 2 gives connection to the $(k-1)$th train on track 6.

Using a similar reasoning, we find that the other depar-

[4] Under nominal operations the kth train on track 1 (e.g., the one that departs from station A at 10.00 a.m.) gives connection to the $(k-1)$th train on track 4 (which has departed from station D at 9.45 a.m.) and *not* to the kth train on track 4 (which will depart from station D at 10.45 a.m.).

[5] In fact it is sufficient to set the value of $x_4(0)$ such that $x_4(0) + a_{41}(k) + 1$ is smaller than 0. Note that the choice $x_4(0) = -\infty$ guarantees that this condition will always hold.

ture times are given by

$$\begin{aligned}
x_3(k) &= \max(x_2(k) + a_{23}(k) + 1, 30 + (k-1)T) \\
x_4(k) &= \max(x_3(k) + a_{34}(k) + 1, \\
&\qquad x_5(k) + a_{54}(k) + 2 - u_{54}(k), 45 + (k-1)T) \\
x_5(k) &= \max(x_1(k) + a_{15}(k) + 2 - u_{15}(k), \\
&\qquad x_6(k-1) + a_{65}(k) + 1, 20 + (k-1)T) \\
x_6(k) &= \max(x_3(k) + a_{36}(k) + 2 - u_{36}(k), \\
&\qquad x_5(k) + a_{56}(k) + 1, 50 + (k-1)T)
\end{aligned}$$

for $k = 1, 2, \ldots$ with $x_j(0) = -\infty$ for $j = 1, 2, \ldots, 6$.

Let us now assume that all travel times are nominal (cf. Table I) except for $a_{12}(1) = a_{15}(1) = 28$ and $a_{12}(2) = a_{15}(2) = 20$. Let $N_\mathrm{c} = 3$, $N_\mathrm{p} = 5$, $\lambda = 0.5$, $\eta = 0.01$, $c_{62}^{\mathrm{broken}}(k) = c_{54}^{\mathrm{broken}}(k) = 15$, $c_{15}^{\mathrm{broken}}(k) = c_{36}^{\mathrm{broken}}(k) = 10$ and use J_{broken}^* as the cost function for the broken connections (with $\alpha = 4$, $\beta = 0.05$ and $\gamma = 0.75$).

If we do not break any connections then we find a maximal delay w.r.t. the departure time schedule of 11 minutes in the first cycle (for the train on track 2), 9 minutes in the second cycle (for the train on track 2), and 4 minutes in the third cycle (for the train on track 1). In the fourth cycle the trains will again ride on schedule. If we do not break any connections, then the value of the total MPC cost function ((8) included) is 87.

If we compute the optimal MPC control input for $k = 1$, we find with both the nonlinear optimization approach and the ELCP approach the following solution: completely break the connection $1 \to 5$ in the first cycle, and partially break connection $3 \to 6$ during the first cycle and connections $1 \to 5$ and $5 \to 4$ during the second cycle[6]. If we apply this control strategy, then we find a maximal delay w.r.t. the departure time schedule of 11 minutes in the first cycle and 5 minutes in the second cycle (both for the train on track 2). In the third cycle all the trains will again ride on schedule. The corresponding value of the total MPC cost function ((8) included) is approximately 46.71.

VI. Conclusions

We have presented an MPC-like control design method for a class of discrete event systems with both soft and hard synchronization constraints. The control action consists in breaking certain soft connections to prevent delays from accumulating, but this can only be done at a certain cost. We have also shown that the resulting optimization problem can be solved using ELCPs. Furthermore, due to the use of a moving horizon strategy and a control horizon this method can be used in on-line applications and it can deal with (predicted) changes in the system parameters. So if we can predict the delays that will occur due to an incident or to works, then we can include this information when determining the optimal control input for the next cycles of the operation of the network.

[6] More specifically, only allow transfer/connection times of $t_{36}(1) = 1.14$, $t_{15}(2) = 0.97$ and $t_{54}(2) = 1.23$ minutes instead of $t_{36}^{\min}(1) = t_{15}^{\min}(2) = t_{54}^{\min}(2) = 2$ minutes.

An important topic for future research is the development of efficient algorithms to solve the railway MPC problem. One option could be to develop a branch-and-bound algorithm to solve optimization problems defined over the solution set of an ELCP. So instead of first determining the solution set of the ELCP (which is a computationally intensive operation) and then optimizing the objective function over the parameterized solution set, we could then perform the optimization and the (implicit) solution of the ELCP in one step, which should lead to a much more efficient approach. We will also compare the performance of this branch-and-bound algorithm with the straightforward nonlinear non-convex optimization approach.

Acknowledgments

Research partially sponsored by the European Community TMR project ALAPEDES (Algebraic Approach to Performance Evaluation of Discrete Event Systems), and by the FWO Research Community ICCoS (Identification and Control of Complex Systems).

References

[1] D.W. Clarke, C. Mohtadi, and P.S. Tuffs, "Generalized predictive control – Part I. The basic algorithm", *Automatica*, vol. 23, no. 2, pp. 137–148, Mar. 1987.

[2] C.E. García, D.M. Prett, and M. Morari, "Model predictive control: Theory and practice – A survey", *Automatica*, vol. 25, no. 3, pp. 335–348, May 1989.

[3] L. Biegler, "Efficient solution of dynamic optimization and NMPC problems", in *Nonlinear Model Predictive Control*, F. Allgöwer and A. Zheng, Eds. 2000, vol. 26 of *Progress in Systems and Control Theory*, Basel, Switzerland: Birkhäuser Verlag.

[4] B. De Schutter and T. van den Boom, "Model predictive control for max-plus-linear discrete event systems", *Automatica*, vol. 37, no. 7, pp. 1049–1056, July 2001.

[5] F. Baccelli, G. Cohen, G.J. Olsder, and J.P. Quadrat, *Synchronization and Linearity*, John Wiley & Sons, New York, 1992.

[6] R.A. Cuninghame-Green, *Minimax Algebra*, vol. 166 of *Lecture Notes in Economics and Mathematical Systems*, Springer-Verlag, Berlin, Germany, 1979.

[7] J.G. Braker, "Max-algebra modelling and analysis of time-table dependent transportation networks", in *Proceedings of the 1st European Control Conference*, Grenoble, France, July 1991, pp. 1831–1836.

[8] J.G. Braker, *Algorithms and Applications in Timed Discrete Event Systems*, PhD thesis, Department of Technical Mathematics and Informatics, Delft University of Technology, Delft, The Netherlands, Dec. 1993.

[9] J.G. Braker, "An extended algorithm for performance evaluation of timed event graphs", in *Proceedings of the 2nd European Control Conference*, Groningen, The Netherlands, June 1993, pp. 524–529.

[10] R. de Vries, B. De Schutter, and B. De Moor, "On max-algebraic models for transportation networks", in *Proceedings of the 4th International Workshop on Discrete Event Systems (WODES'98)*, Cagliari, Italy, Aug. 1998, pp. 457–462.

[11] P.R. de Waal, A. Overkamp, and J.H. van Schuppen, "Control of railway traffic on a single line", in *Proceedings of the European Control Conference (ECC'97)*, Brussels, Belgium, paper 230, July 1997.

[12] R. Minciardi, M. Paolucci, and R. Pesenti, "Generating optimal schedules for an underground railway line", in *Proceedings of the 34th IEEE Conference on Decision and Control*, New Orleans, Louisiana, Dec. 1995, pp. 4082–4085.

[13] B. De Schutter and B. De Moor, "The extended linear complementarity problem", *Mathematical Programming*, vol. 71, no. 3, pp. 289–325, Dec. 1995.